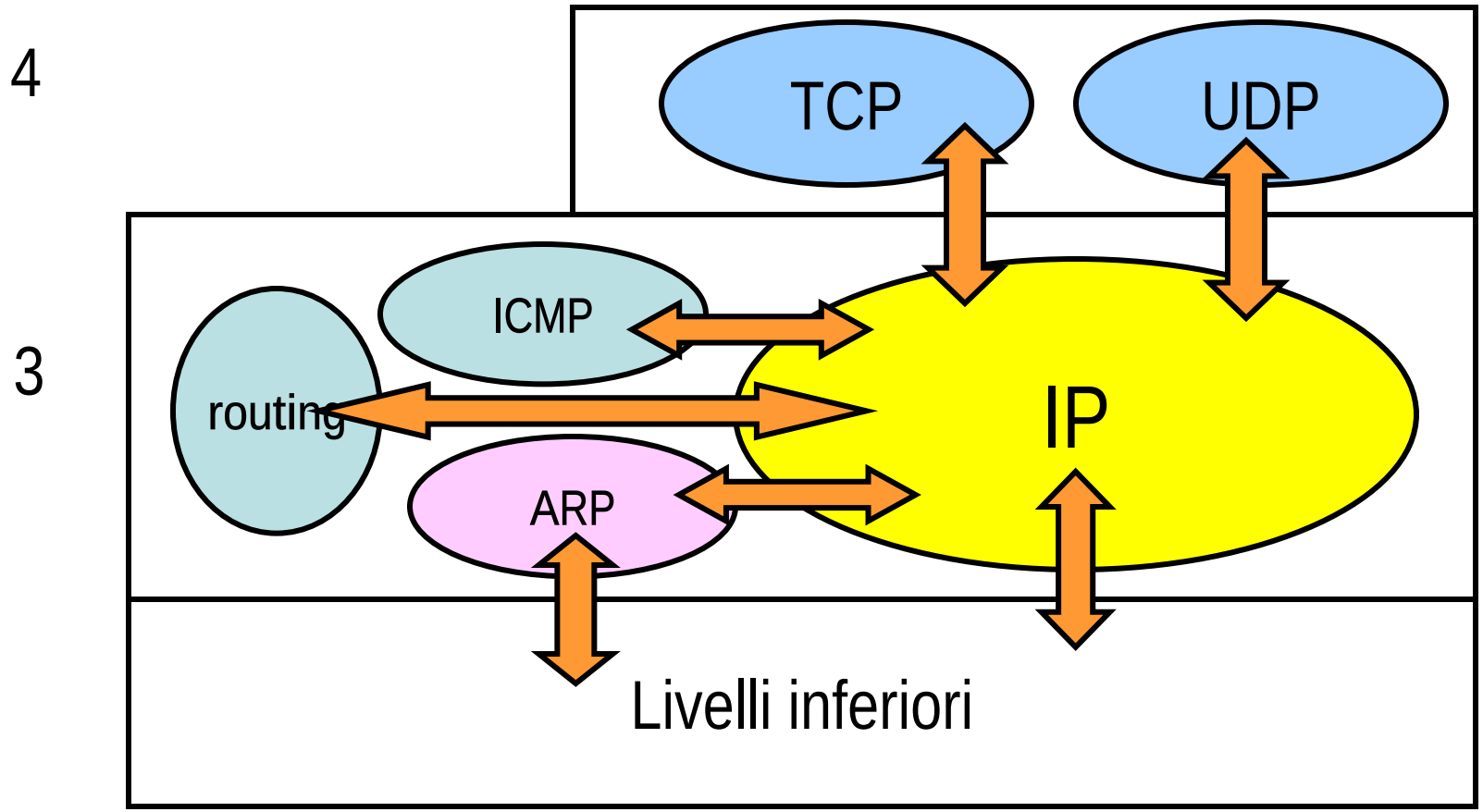
Università di Bergamo

Dipartimento di Ingegneria dell'Informazione e Metodi Matematici

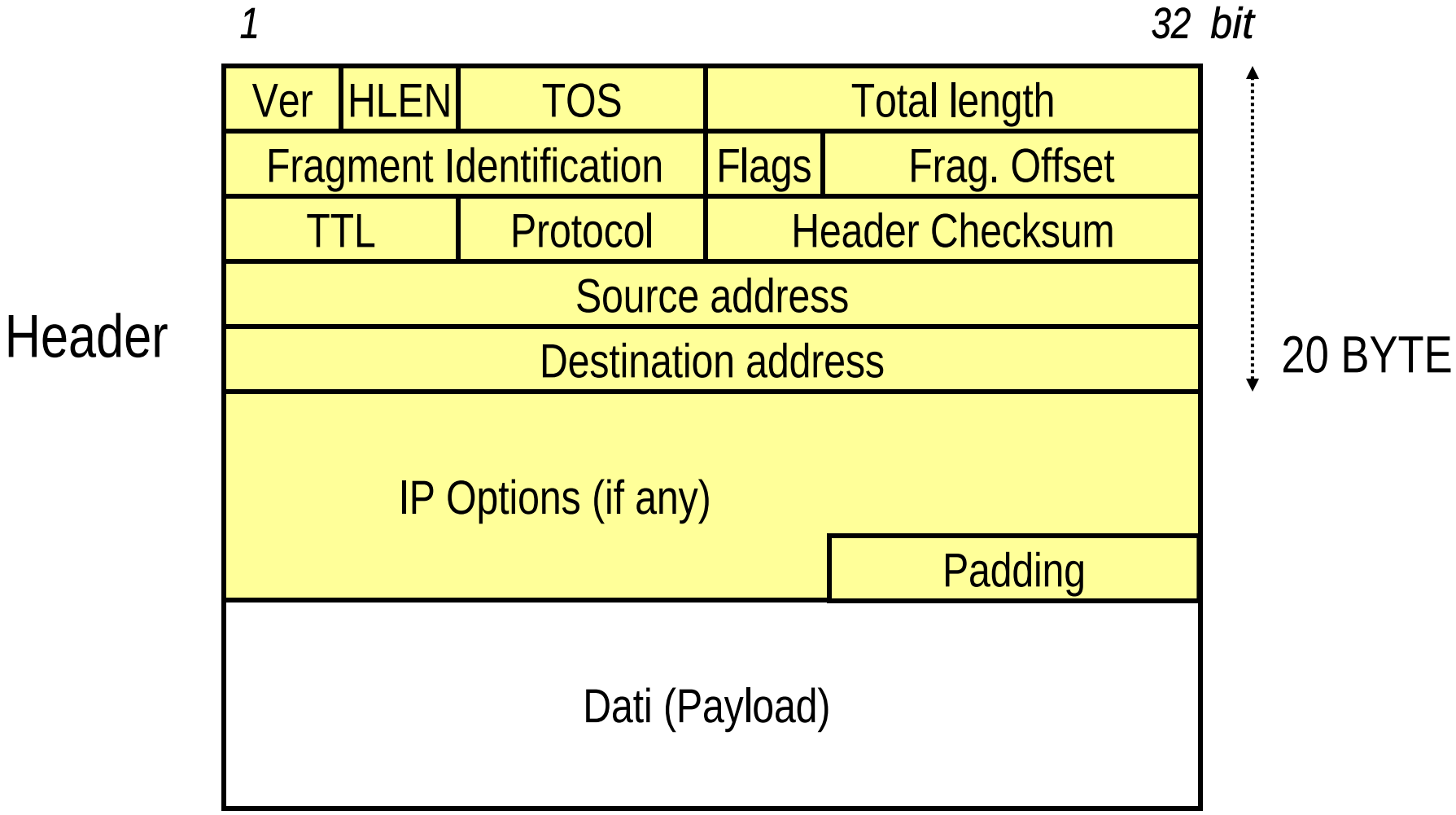
Fondamenti di Reti e Telecomunicazione

Internet Protocol

Lo stack TCP/IP base



Il pacchetto/datagramma IP (RFC 791)



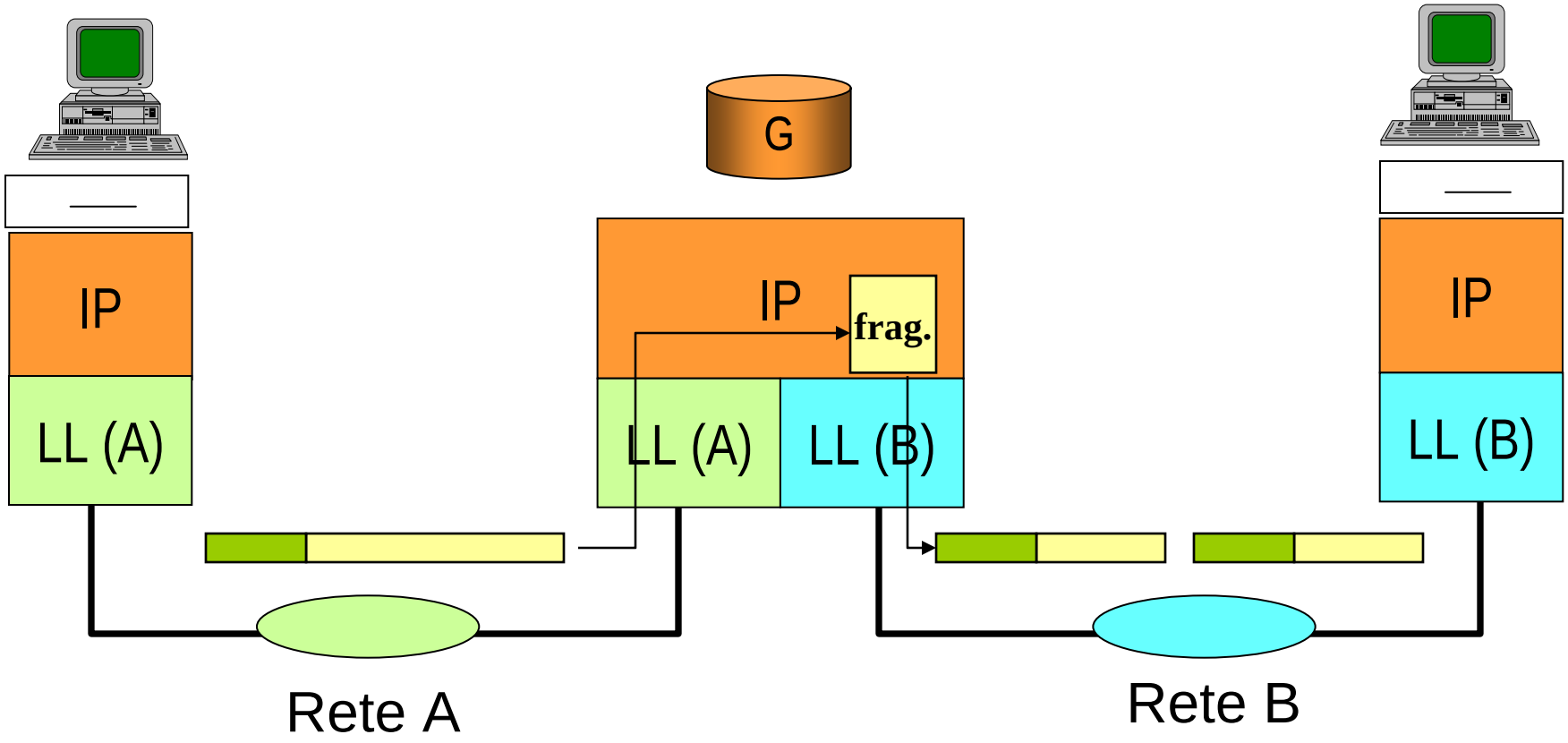
Il pacchetto IP

- **Ver (4 bit):**
  - **version: indica la versione del protocollo; quella che noi studiamo è la versione 4**
- **HLEN (4 bit)**
  - **header length: indica la lunghezza dell'*header* del pacchetto (comprese opzioni e padding) espressa in parole da 32 bit (4 byte). Minimo valore valido: 5**
- **TOS (8 bit)**
  - **Type Of Service: un campo che adesso prende il nome di DS field (RFC 2474) e può essere utilizzato per la gestione delle priorità nelle code dei router**
- **Total length (16 bit):**
  - **indica la lunghezza totale del pacchetto in byte: valore massimo 2^{16}=65536; una volta sottratta la dimensione dell'header, si ha la lunghezza del payload**

La frammentazione

- **Identification, Flags, Fragment Offset**
  - **Alcuni protocolli di livello inferiore a cui IP si appoggia richiedono una dimensione massima del pacchetto (MTU) inferiore a 65536 bytes (tipico l'esempio di Ethernet che accetta pacchetti fino a 1500 bytes)**
  - **prima di passare il pacchetto al livello inferiore, IP divide il pacchetto in frammenti, ciascuno con il proprio header**
  - **i frammenti verranno ricomposti dall'entità IP del destinatario**
  - **i campi Identification, Flags e Frag. Offset sono usati per questo scopo**

La frammentazione

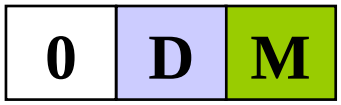


La frammentazione

- **Identification (16 bit)**
  - **è un campo che identifica tutti i frammenti di uno stesso pacchetto in modo univoco. E' scelto dall'IP Sender**
- **Frag. Offset (13 bit)**
  - **I byte del pacchetto originale sono numerati da 0 al valore della lunghezza totale. Il campo Frag. Offset identifica la posizione del frammento nel datagramma IP originale (in multipli di 8 byte). Il primo frammento ha Offset pari a 0.**
  - **Ad esempio: se un pacchetto di 2000 byte viene diviso in due da 1000 il primo frammento avrà un offset pari a 0 e il secondo pari a 1000 (ovvero: nel campo Frag. Offset del secondo troveremo scritto 1000/8=125)**

La frammentazione

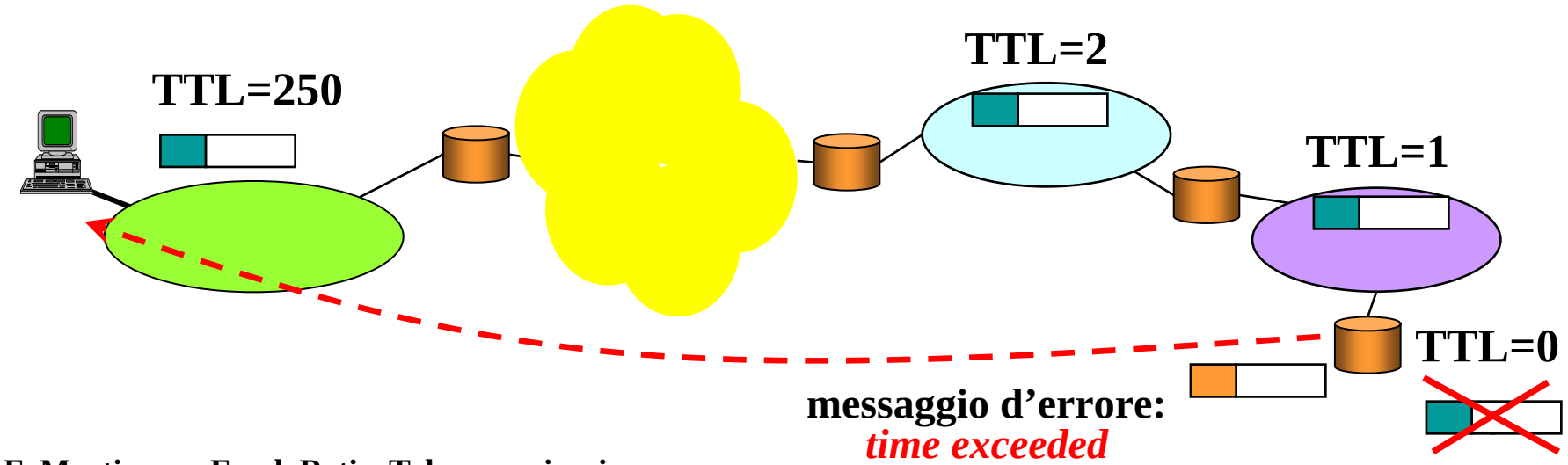
- **Flags (3 bit)**



  - **Il primo bit è riservato e deve contenere 0**
  - **il bit M (More) è pari a 0 solo nell'ultimo frammento (*last fragment*), ad 1 negli altri (*more fragments*)**
  - **il bit D viene posto a 1 quando non si vuole che lungo il percorso venga applicata la frammentazione**
    - ✓**in questo caso se la frammentazione fosse necessaria non viene applicata ma viene generato un messaggio di errore**

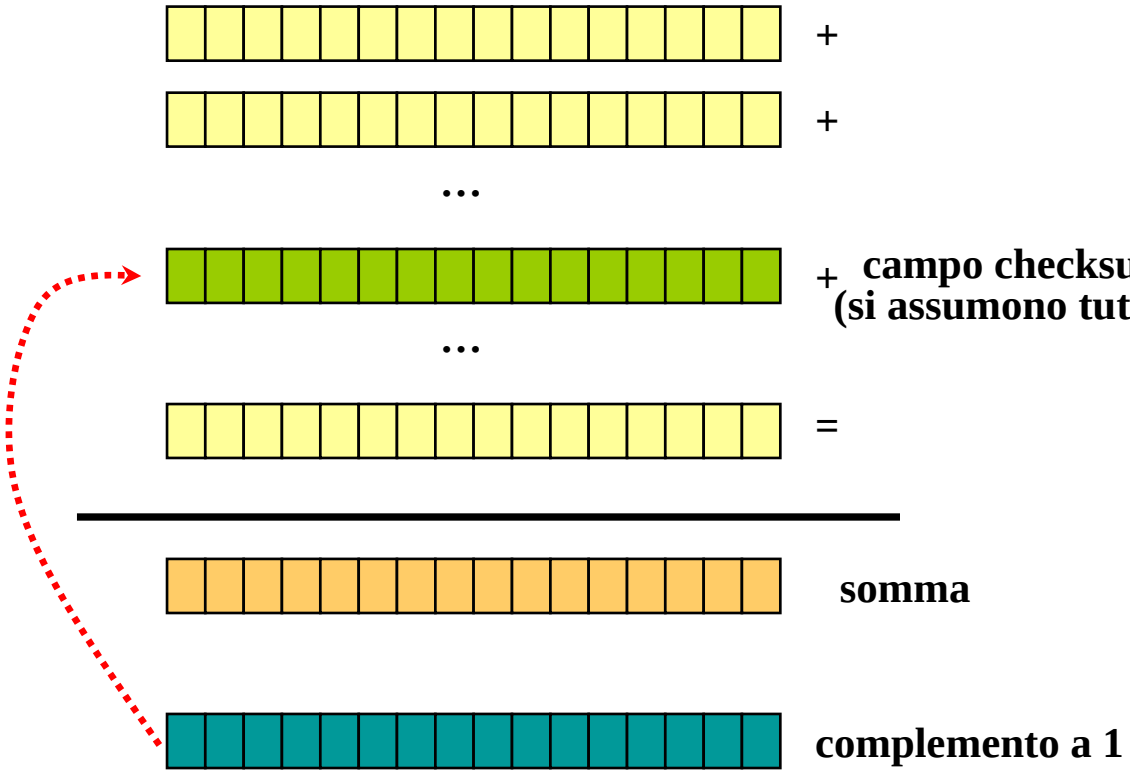
TTL (Time To Live) (8 bit)

- Il campo TTL viene settato ad un valore elevato da chi genera il pacchetto e viene decrementato da ogni router attraversato
- Se un router decrementa il valore e questo va a zero, il pacchetto viene scartato e viene generato un messaggio di errore verso la sorgente



Checksum (16 bit)

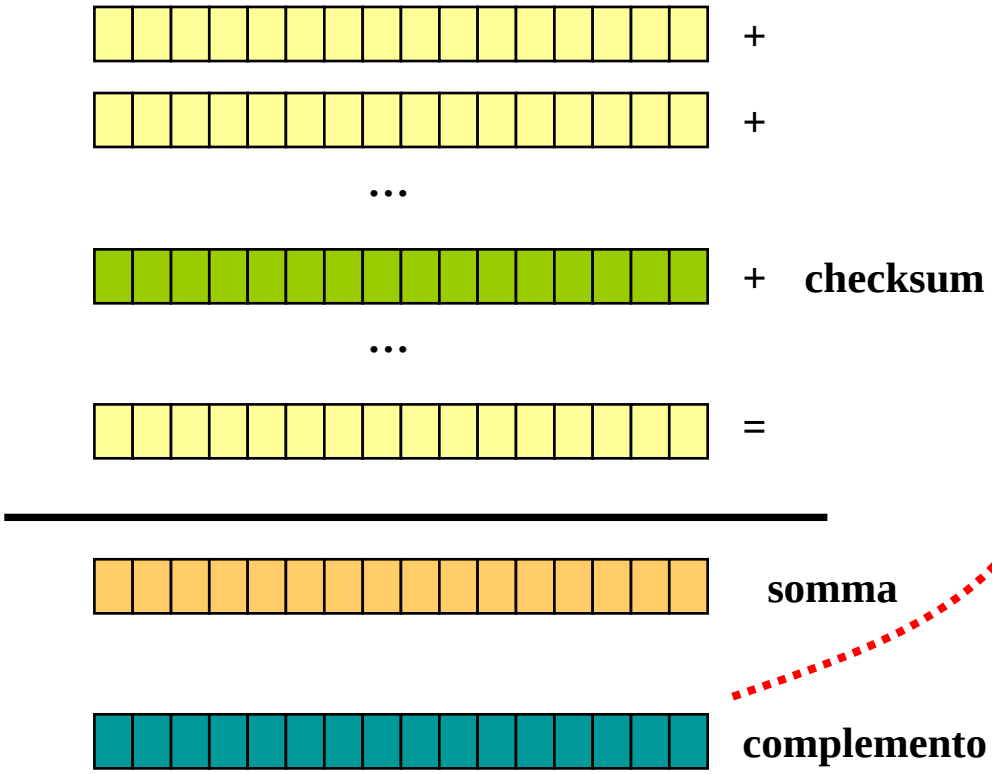
- serve per individuare eventuali errori nell'header (e solo nell'header)
- viene calcolato dal mittente e controllato dal destinatario (ad ogni hop)



- l'header viene diviso in blocchi di 16 bit
- viene fatta la somma modulo 2 dei bit corrispondenti in ciascun blocco
- il risultato viene complementato e quindi inserito nel campo checksum

Checksum

- In ricezione si calcola la somma e si verifica il complemento:



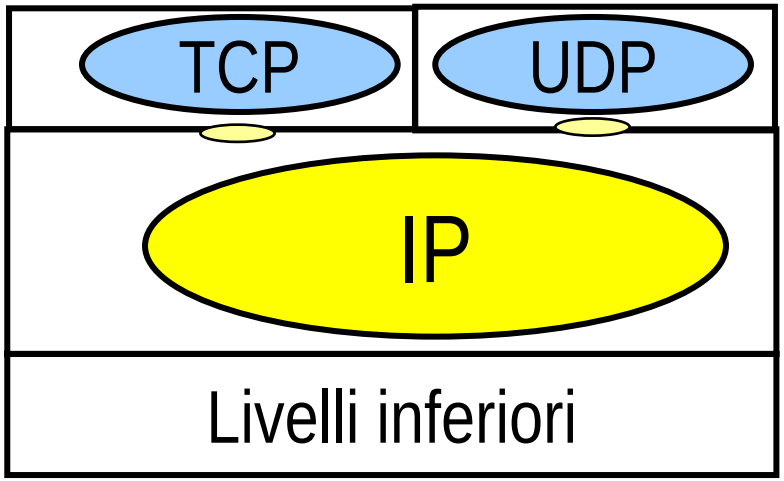
- se sono tutti 0 il pacchetto viene accettato
- altrimenti viene scartato!

Checksum

- **Nota: poiché esistono campi dell'header IP che cambiano a mano a mano che il pacchetto viene inoltrato (es. Time To Live, TTL), ogni entità IP lungo il percorso ricalcola il checksum**
- **L'entità IP del nodo successivo può così verificare l'integrità dell'header ed accettare o meno il pacchetto IP**

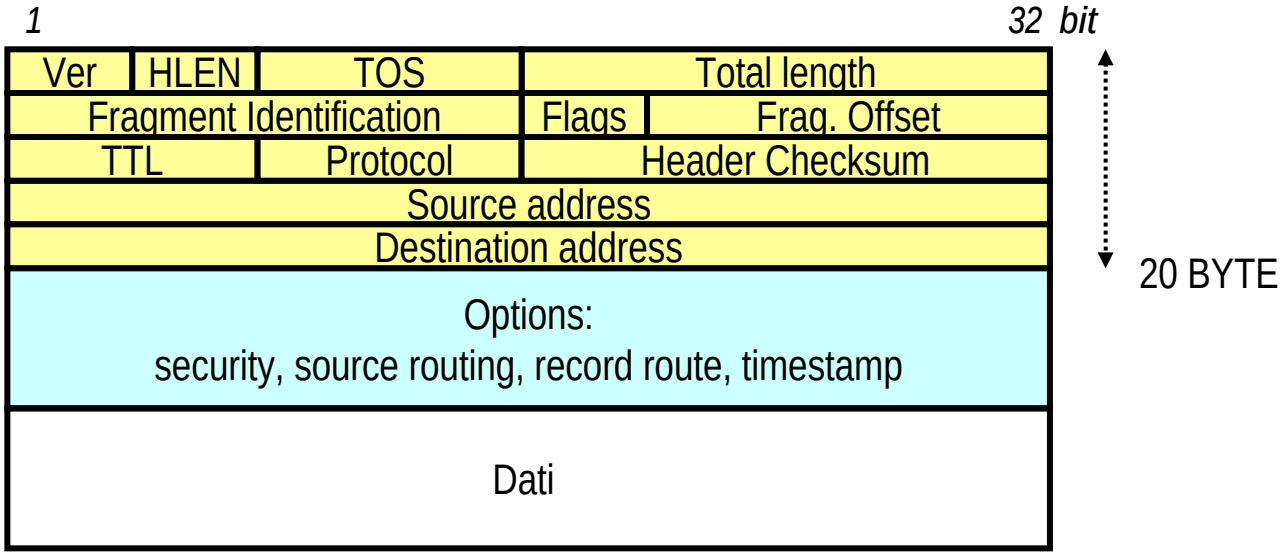
Protocol (8 bit)

- E’ un codice che indica il protocollo di livello superiore (RFC 790)
- Esempio: ICMP=1, TCP=6 …
- più protocolli di livello superiore possono usare IP (multiplazione)
- il codice identifica il SAP (Service Access Point) tra IP e il protocollo di livello superiore

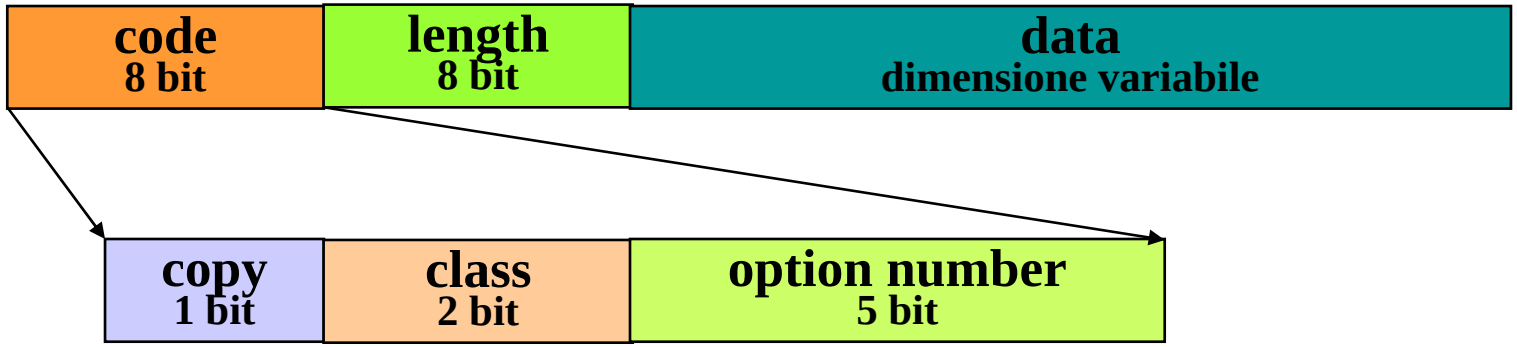


Le opzioni

- La parte iniziale dell'header IP è di 20 byte ed è sempre presente
- In aggiunta è possibile la presenza di campi aggiuntivi (le opzioni) che possono allungare l'header fino ad un massimo di 60 byte



Le opzioni



Copy:
- 0 nel primo o unico frammento
- 1 negli altri (copied)

Class:
- 00 controllo del datagram
- 10 debugging and measurement

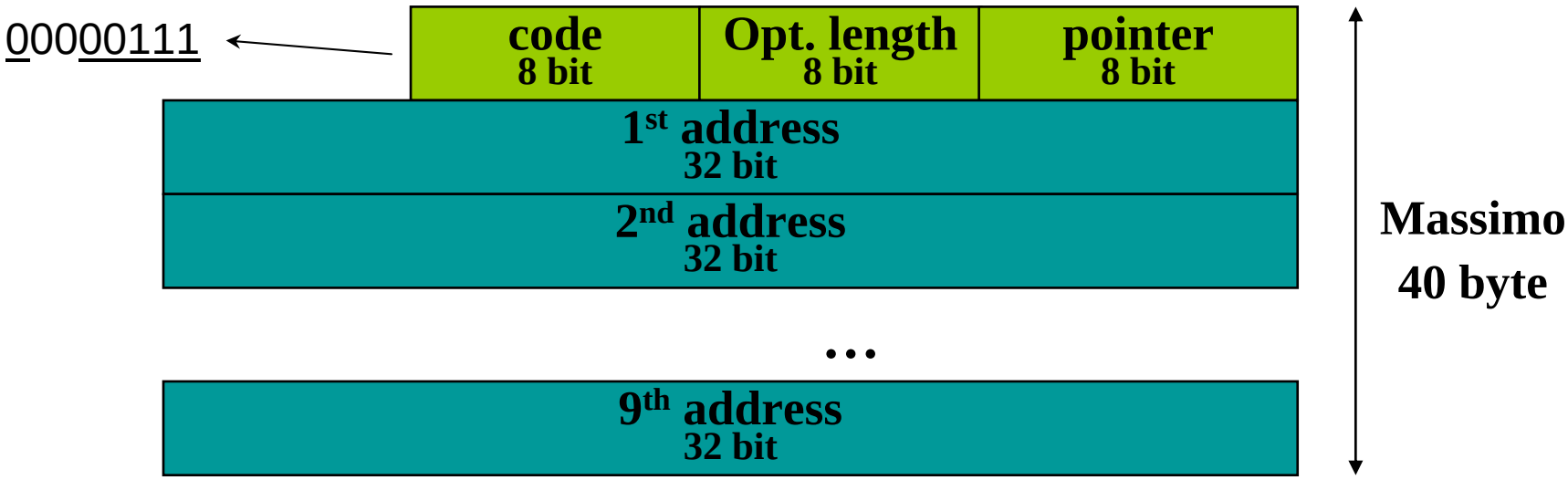
01 e 11 sono riservate per usi futuri

Option number:
- 00000 end of option (1 byte)
- 00001 no operation (1 byte)
- 00011 loose source routing
- 00100 time stamp
- 00111 record route
- 01001 strict source routing

Le opzioni

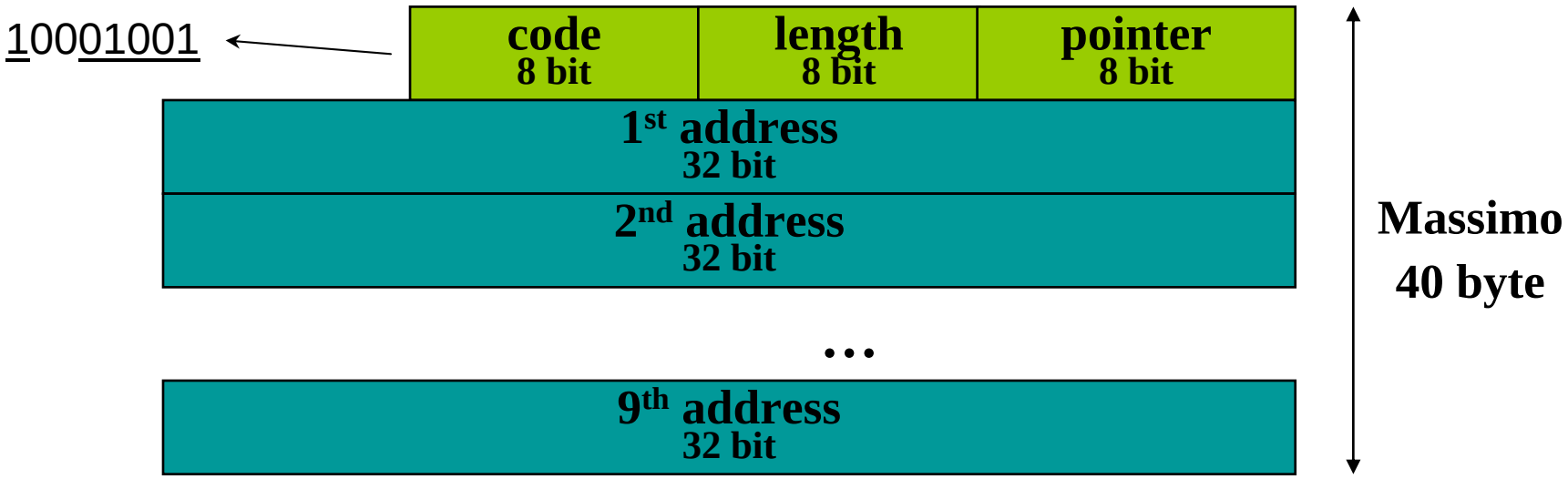
CLASS	NUMBER	LENGTH	DESCRIPTION
0	0	-	End of Option list. This option occupies only 1 octet; it has no length octet.
0	1	-	No Operation. This option occupies only 1 octet; it has no length octet.
0	2	11	Security. Used to carry Security, Compartmentation, User Group (TCC), and Handling Restriction Codes compatible with DOD requirements.
0	3	var.	Loose Source Routing. Used to route the internet datagram based on information supplied by the source.
0	9	var.	Strict Source Routing. Used to route the internet datagram based on information supplied by the source.
0	7	var.	Record Route. Used to trace the route an internet datagram takes.
0	8	4	Stream ID. Used to carry the stream identifier.
2	4	var.	Internet Timestamp.

Record Route



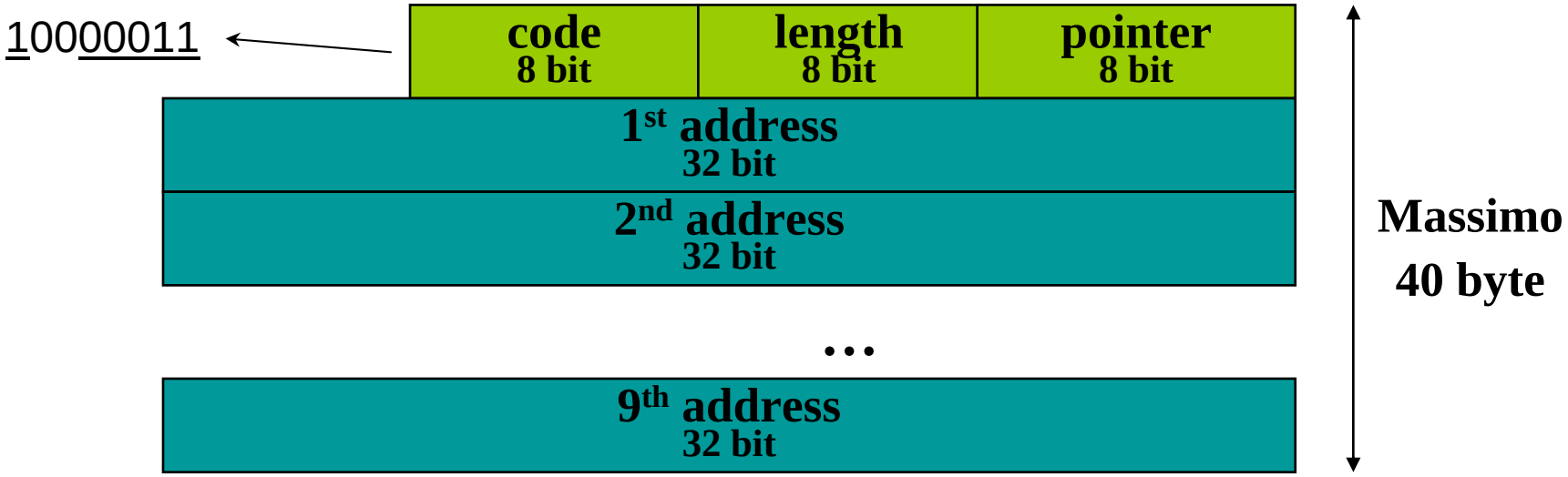
- Il pointer indica l'ottetto con cui comincia la prossima area in cui registrare un indirizzo. Il puntatore è relativo a questa opzione.
- Tutti i campi address sono inizialmente vuoti e il pointer vale 4 (ovvero punta al primo campo address, che comincia appunto al 4º ottetto dall'inizio dell'opzione)
- ogni volta che viene attraversato un router viene registrato l'indirizzo nel campo puntato e il puntatore viene aumentato di 4, fino all'eventuale riempimento di tutti i campi address
- (per conoscere il percorso verso una destinazione esiste la possibilità di usare pacchetti ICMP come vedremo in seguito)

Strict Source Routing



- Implementa un meccanismo di source routing (percorso scelto dalla sorgente)
- Tutti i campi address sono inizialmente pieni e indicano i router che l'IP sender vuole vengano attraversati
- il puntatore viene incrementato di 4 ad ogni hop
- se viene raggiunto un router non previsto il pacchetto viene scartato e viene generato un messaggio di errore
- (usata molto raramente!!!)

Loose Source Routing



- come la precedente, ma è possibile visitare anche altri router (il pacchetto non viene scartato)
- (usata molto raramente!!!)

Timestamp

01000100 ←

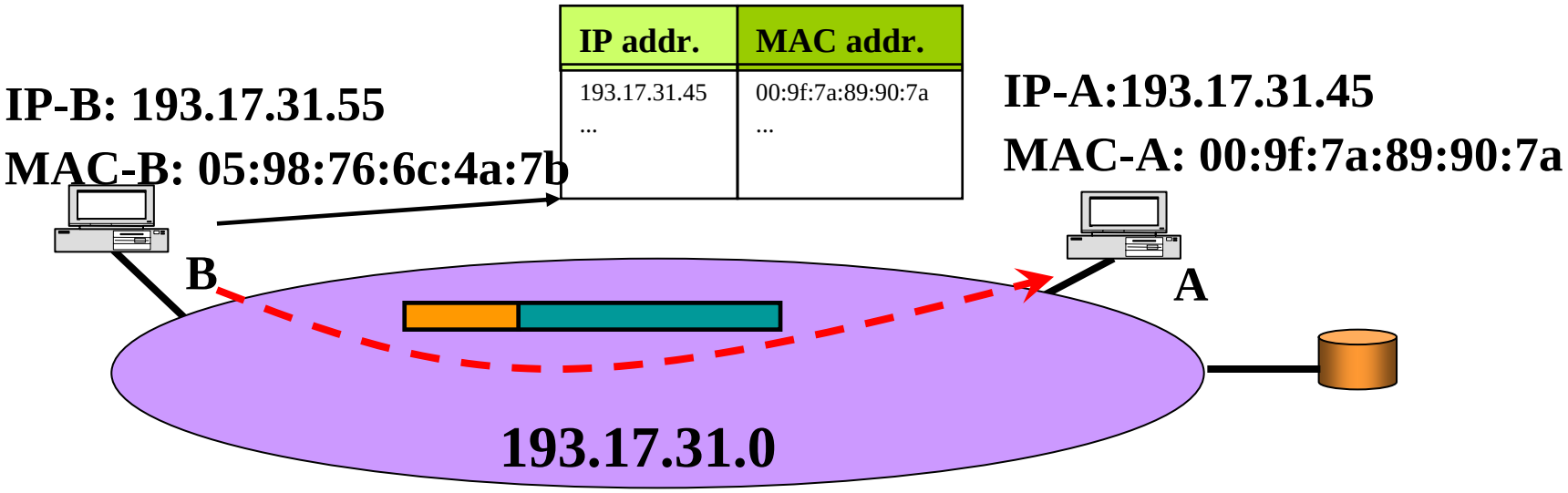
code 8 bit	length 8 bit	pointer 8 bit	O-Flow 4 bit	Flag 4 bit
1st address 32 bit				
1st time stamp 32 bit				
2nd address 32 bit				
2nd time stamp 32 bit				

…

- misura il tempo assoluto di uscita del pacchetto in un router
- il campo Over-Flow indica il numero di router sul percorso che non hanno potuto aggiungere il timestamp (per mancanza di spazio nell'opzione, che al massimo può raggiungere i 40 byte)
- il campo Flag indica la modalità operativa stabilita dal mittente (address riempiti dal mittente o dai router, ecc.)

Corrispondenza tra indirizzi IP e indirizzi fisici

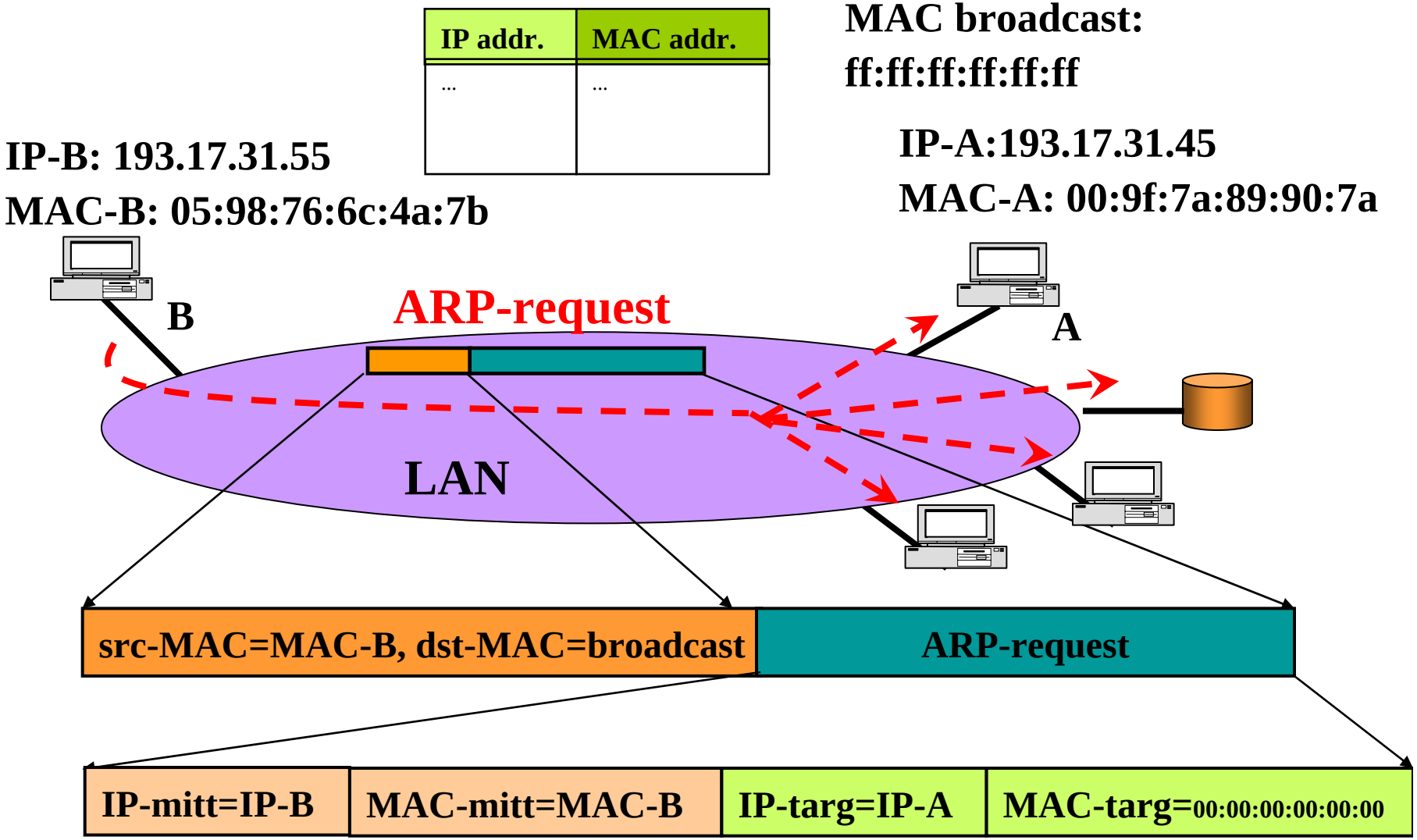
- Illustrando le tecniche di inoltro abbiamo ipotizzato la presenza di una tabella di corrispondenza tra indirizzi IP e indirizzi di livello inferiore (indirizzi fisici)
- Queste tabelle vengono create dinamicamente da ciascun host mediante il protocollo ARP



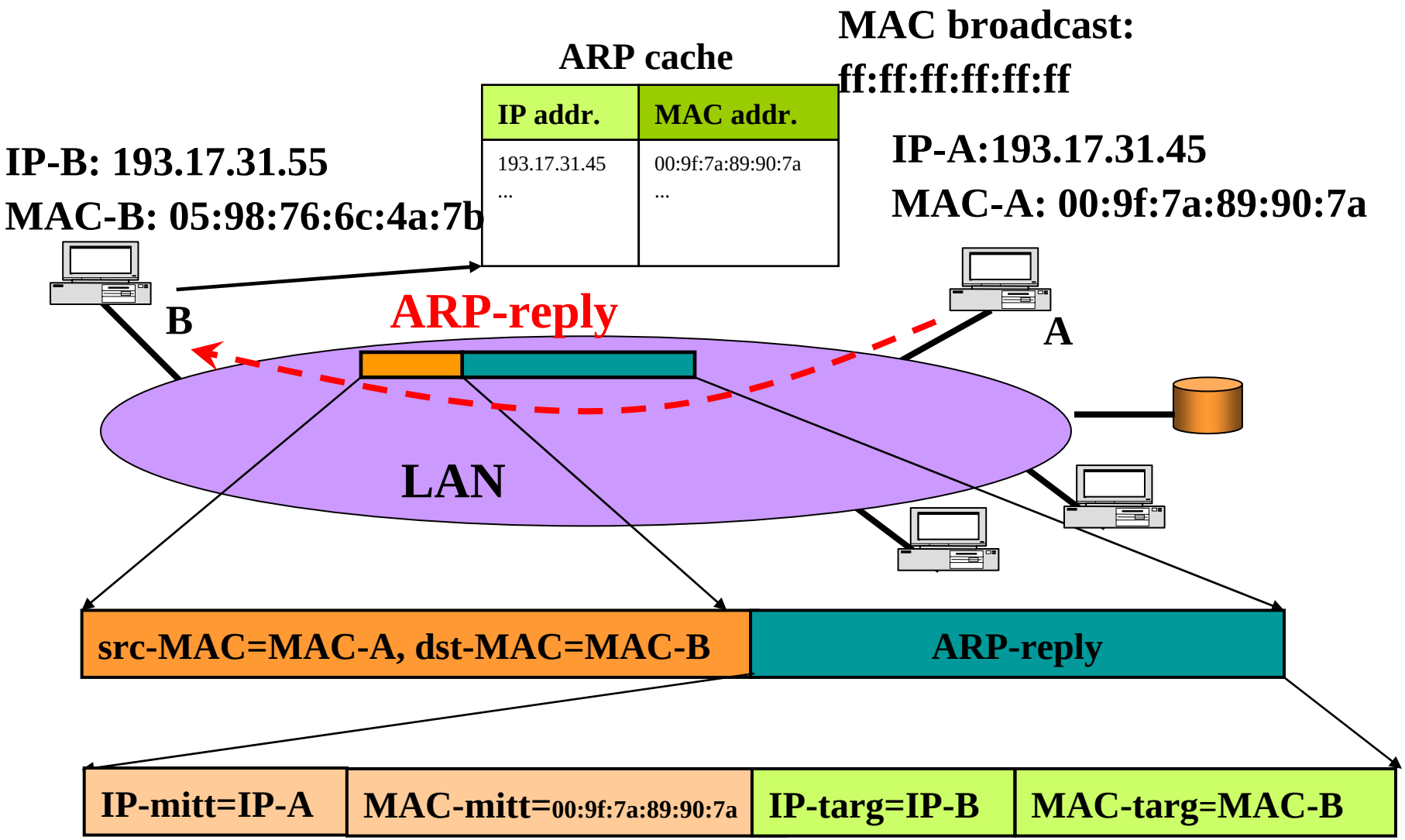
ARP (Address Resolution Protocol)

- Il meccanismo si basa sulla capacità di indirizzamento broadcast della rete locale
- quando nella tabella memorizzata nell'host (denominata *ARP-cache*) non è presente l'indirizzo cercato, viene generato un messaggio di ARP-request
- La ARP-request viene inviata in broadcast e contiene l'indirizzo IP di cui si chiede il corrispondente indirizzo MAC
- L'host che riconosce l'indirizzo IP come proprio invia una ARP-reply direttamente a chi aveva inviato la richiesta, con l'indicazione dell'indirizzo MAC

ARP (Address Resolution Protocol)



ARP (Address Resolution Protocol)



Formato dei pacchetti ARP

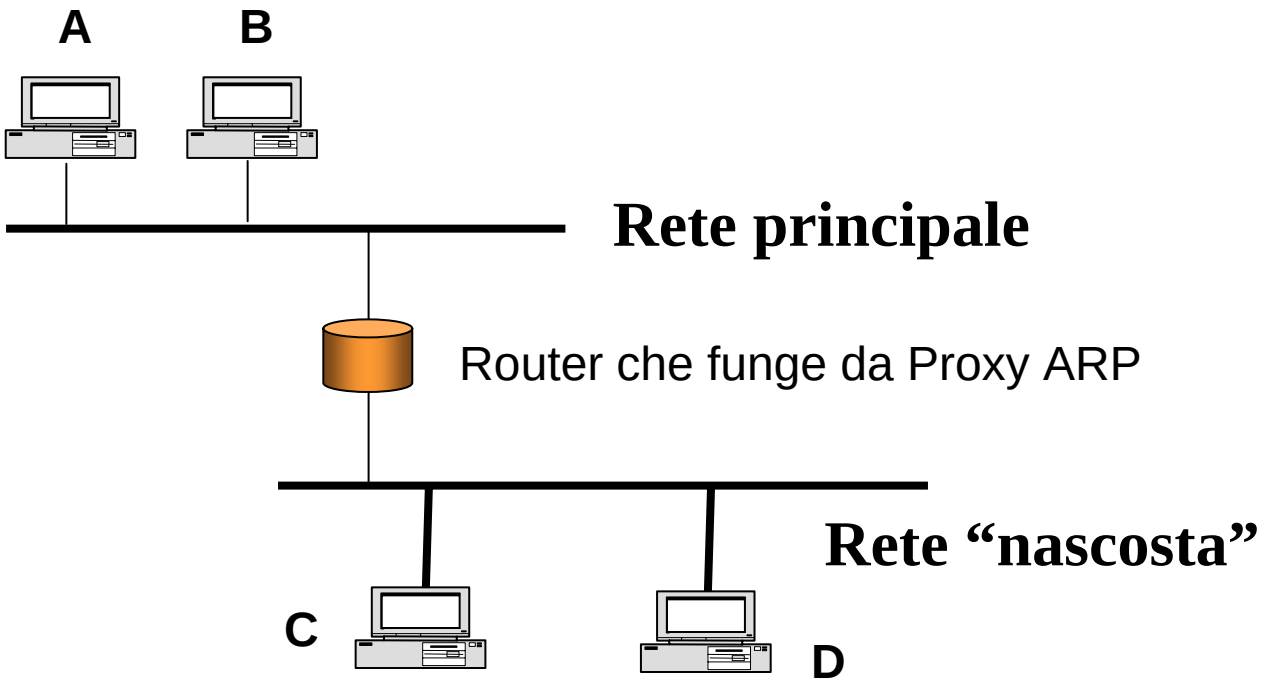
- **ARP può essere usato per altri protocolli di livello 2 e livello 3 quindi occorre indicare il tipo di protocollo (IP nel nostro caso) e il tipo di hardware (Ethernet per esempio)**
- **Ovviamente: il formato di un pacchetto ARP (ovvero la lunghezza dei suoi campi) varia in funzione del tipo di hardware e di protocollo utilizzati!**

Domini di broadcast e reti IP

- **Per il funzionamento del meccanismo di inoltro e dell'ARP abbiamo fin qui ipotizzato che una sottorete IP corrisponda uno a uno con una rete locale (Dominio di Broadcast)**
- **In realtà un'unica rete locale può corrispondere a diverse sottoreti IP (per es. perché la numerazione disponibile per una non è sufficiente)**
- **Non è possibile che più reti locali possano coesistere in un'unica sottorete IP perché non potrebbero comunicare**

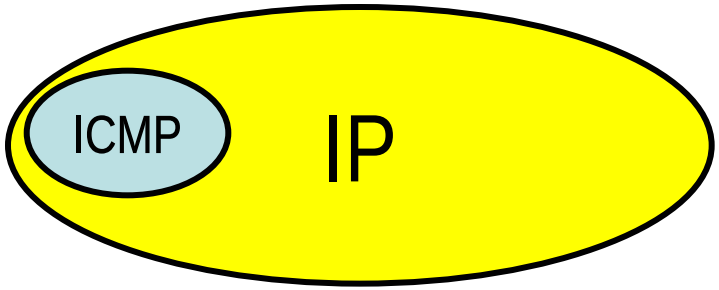
Domini di broadcast e reti IP: proxy ARP

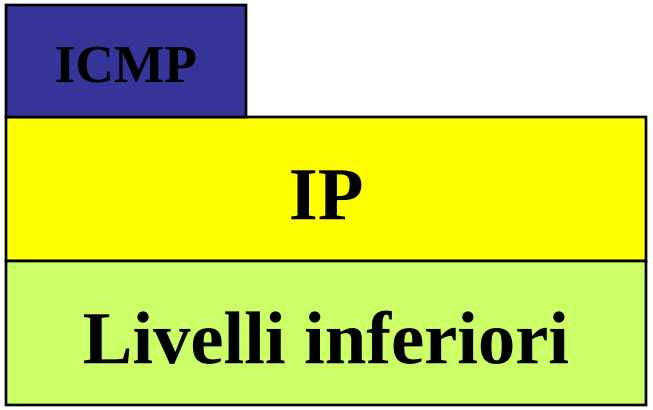
- Un'alternativa è quella dell'installazione di un *proxy ARP* nel router
- La tecnica del proxy ARP consente a due reti fisicamente distinte di condividere lo stesso indirizzo di rete
- Il router conosce la collocazione fisica dei vari host nelle due reti
- Il router risponde alle richieste ARP su ciascuna delle due reti, "fingendosi" il destinatario. Dopodiché instrada i pacchetti al vero host destinatario



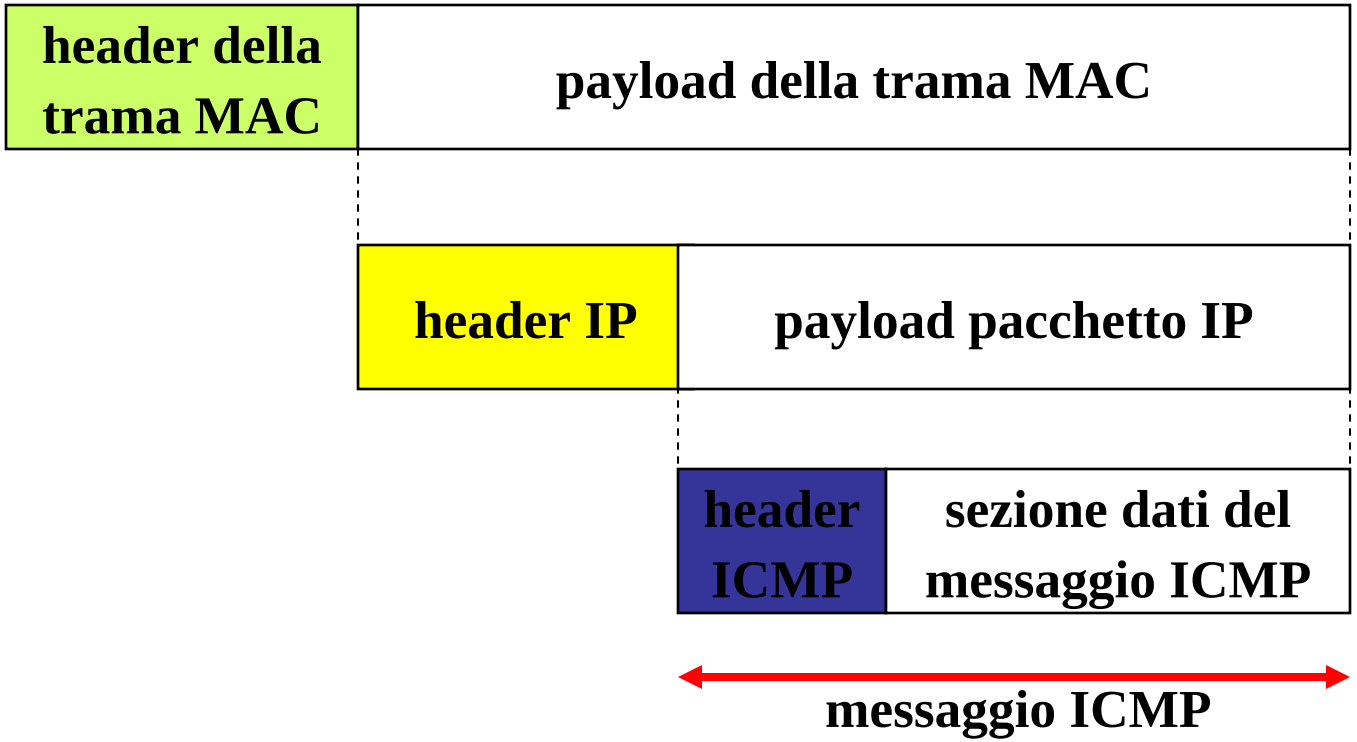
Internet Control Message Protocol (ICMP)

- E’ un protocollo per messaggi di servizio fra host e router, per informazioni su errori e fasi di attraversamento della rete
- da questo punto di vista può essere considerato come parte di IP
- i messaggi ICMP sono incapsulati e trasportati da IP, e quindi da questo punto di vista può essere considerato un utente di IP



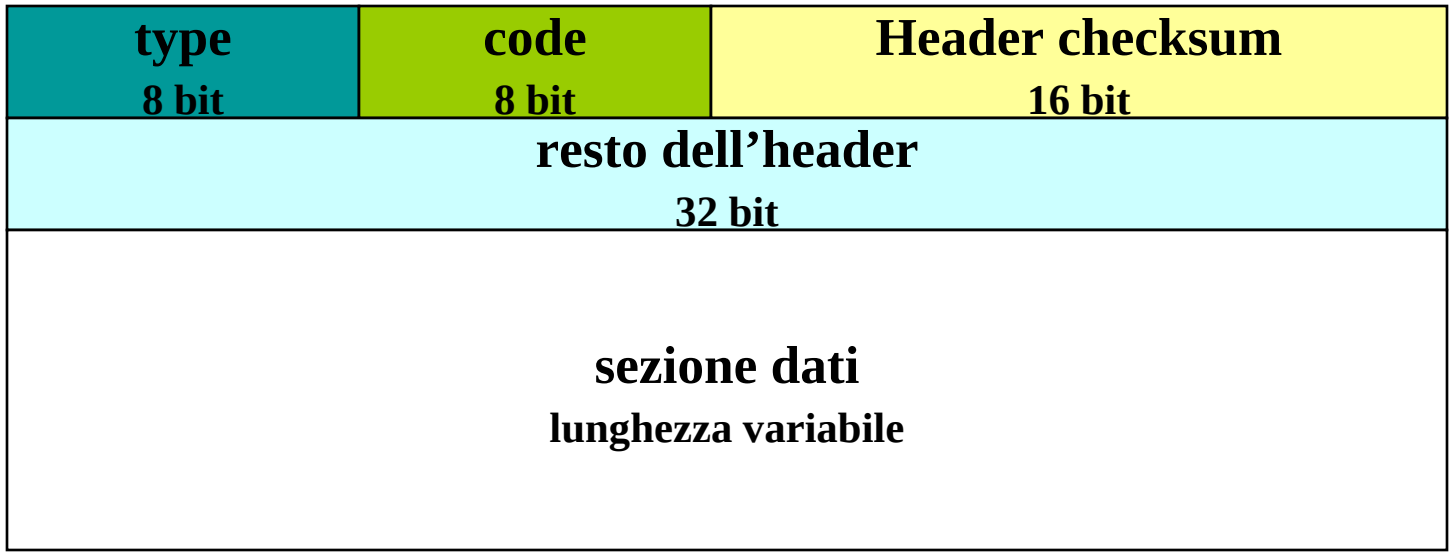


Internet Control Message Protocol (ICMP)



- Nel pacchetto IP il campo protocol indica il codice dell'ICMP (pari ad 1)
- il messaggio ICMP viaggia all'interno del pacchetto IP

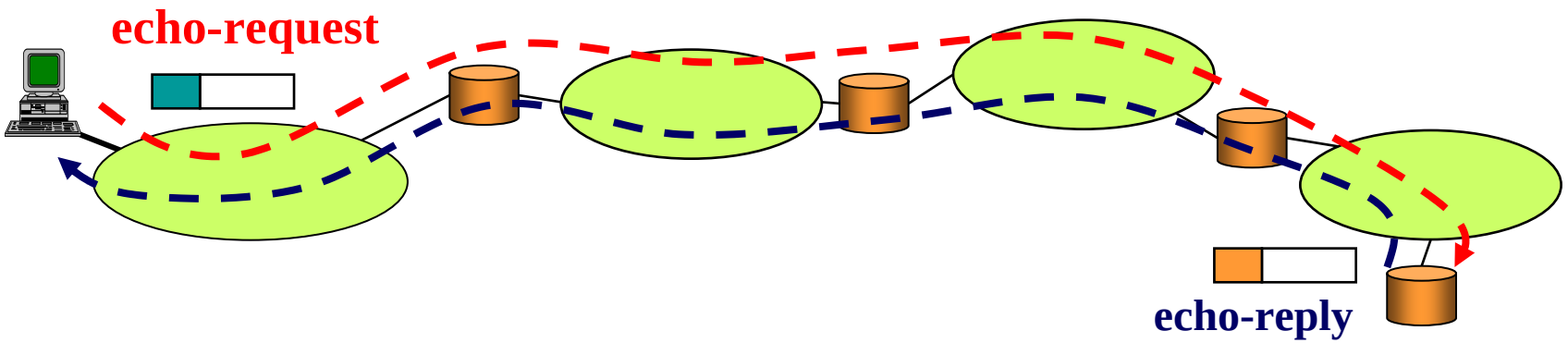
Messaggi ICMP



Type		Type	
0	Echo reply	12	Parameter problem
3	Destination unreachable	13	Timestamp request
4	Source Quench	14	Timestamp reply
5	Redirect (change a route)	17	Address mask request
8	Echo request	18	Address mask reply
11	Time exceeded		

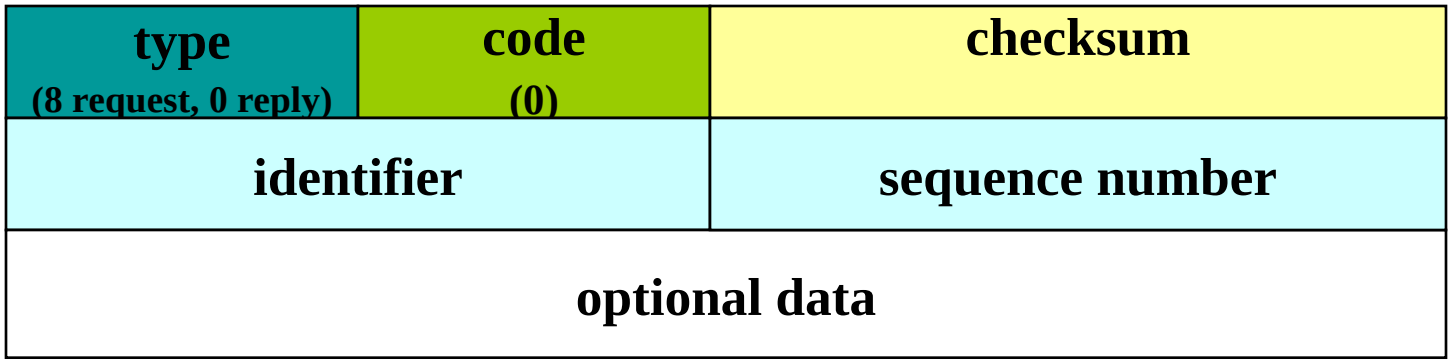
Echo

- I messaggi di Echo-request e Echo-reply sono usati per verificare la raggiungibilità e lo stato di un host o di un router
- quando un nodo IP riceve un messaggio di Echo-request risponde immediatamente con un messaggio di Echo reply



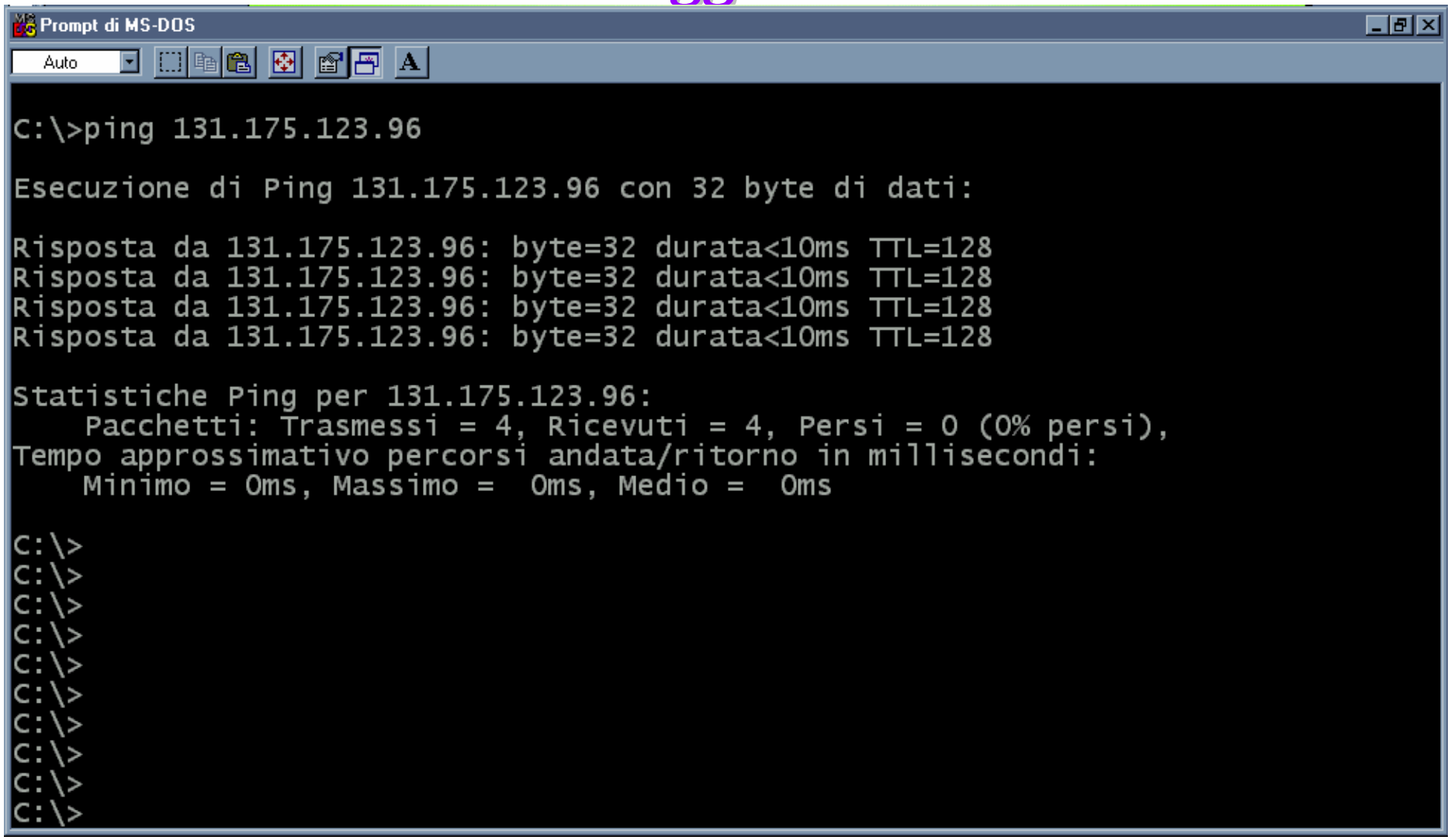
Echo

Protegge solo l'header ICMP
Stesso algoritmo di calcolo usato in IP

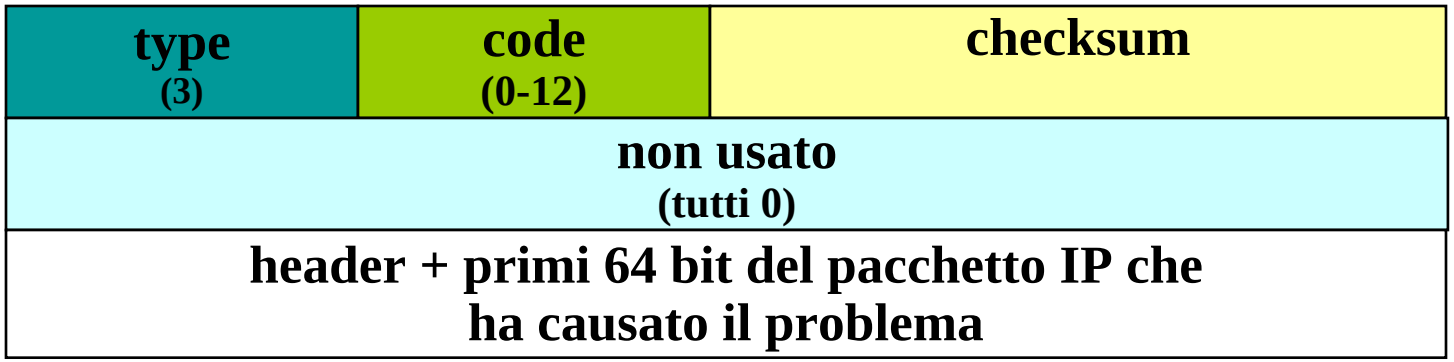


- Il campo identifier viene scelto dal mittente della richiesta
- nella risposta viene ripetuto lo stesso identifier della richiesta
- più richieste consecutive possono avere lo stesso identifier e differire per il sequence number
- una sequenza arbitraria può essere aggiunta dal mittente nel campo optional data e deve essere riportata identica nella risposta

Uso dei messaggi di echo: PING

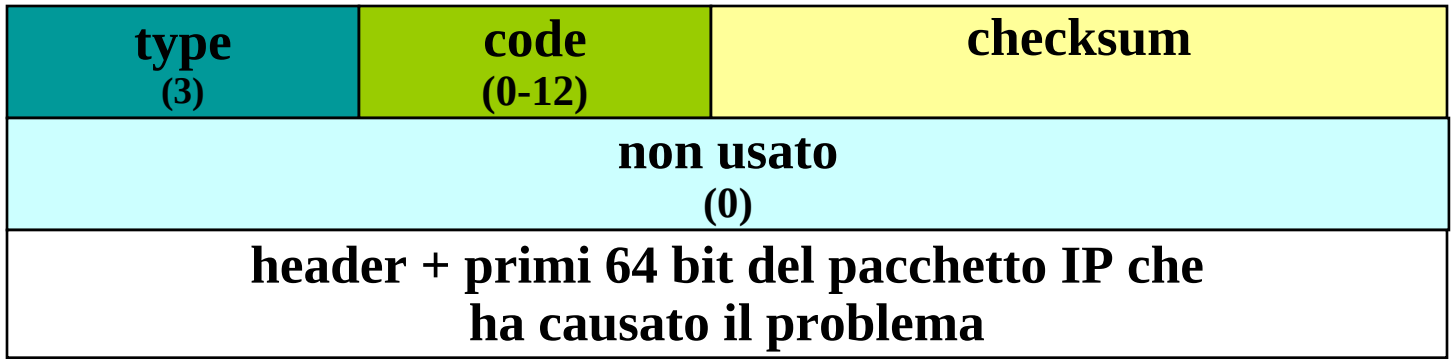


Destination unreachable



- Quando un router scarta un pacchetto per qualche motivo normalmente genera un messaggio di errore che invia alla *sorgente* del pacchetto (IP sender)
- nel campo “code” è codificato il motivo che ha causato l’errore
- ovviamente la generazione del messaggio avviene solo nei casi in cui il router può accorgersi del problema

Destination unreachable



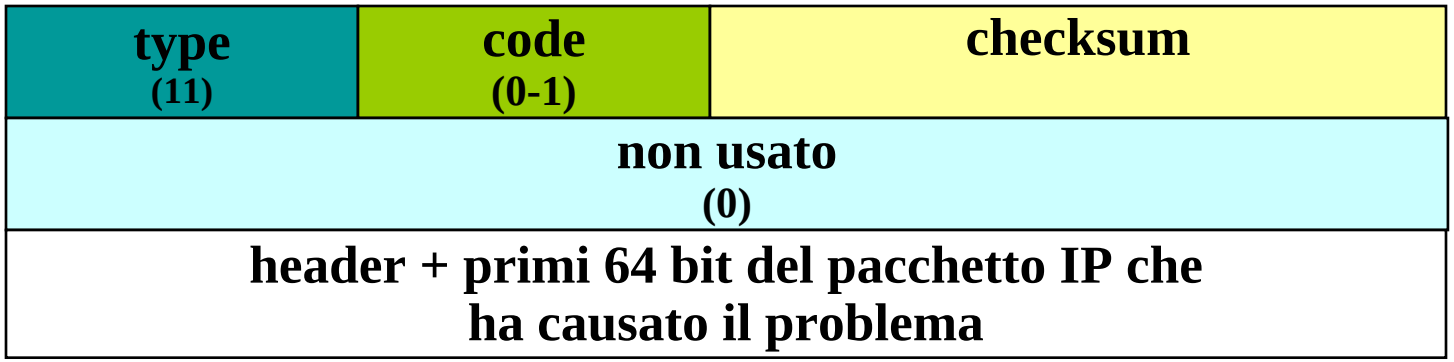
Code possibili:

- **0** network unreachable → Per es. la distanza verso la rete di destinazione è infinita
- **1** host unreachable
- **2** protocol unreachable
- **3** port unreachable

→ Per es. nell'host di destinazione il modulo IP non può consegnare il datagramma perché il modulo del protocollo indicato o la porta del processo non sono attivi

- **4** fragmentation needed and DF set
- **5** source route failed

Time exceeded



- Code 0
  - Il messaggio di time exceeded viene usato quando il router decrementando il TTL lo pone a 0
  - il messaggio di time exceeded viene inviato alla sorgente del pacchetto
- Code 1
  - viene usato dalla destinazione quando non tutti i frammenti di un pacchetto arrivano entro un tempo massimo

Parameter problem

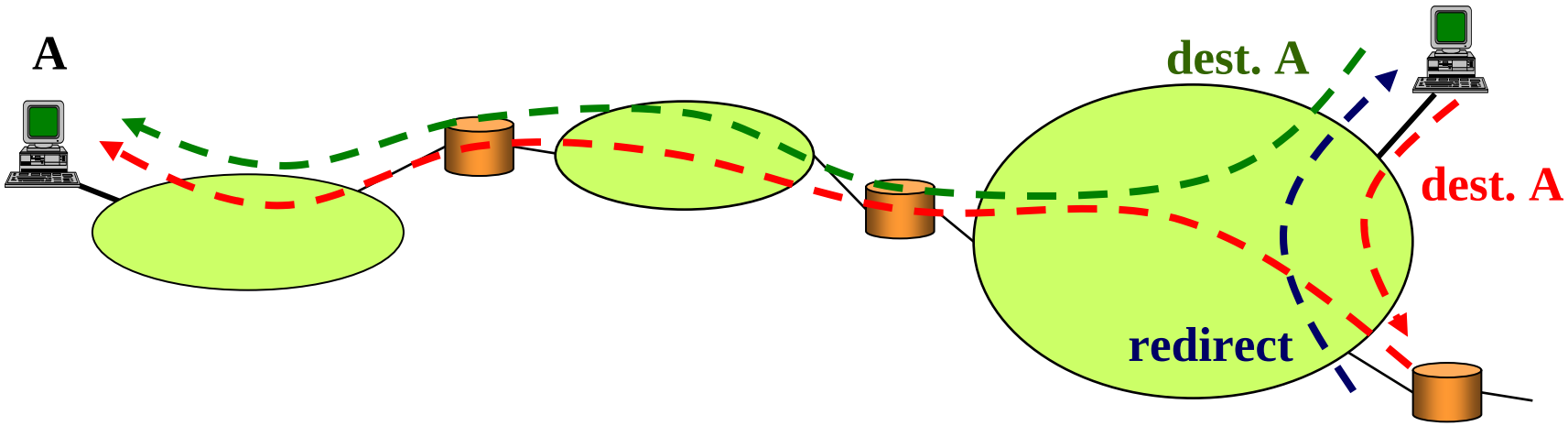
type (12)	code (0-1)	checksum
pointer	non usato (0)	
header + primi 64 bit del pacchetto IP che ha causato il problema		

- Code 0
  - se l'header di un pacchetto IP ha una incongruenza in qualcuno dei suoi campi viene inviato il messaggio di parameter problem; il campo pointer punta al byte del pacchetto che ha causato il problema
- Code 1
  - viene usato quando un'opzione non è implementata e non può essere soddisfatta

Redirect

type (5)	code (0-3)	checksum
indirizzo IP del router		
header + primi 64 bit del pacchetto IP		

- Questo messaggio viene usato quando si vuole che la *sorgente* usi per quella destinazione un diverso router

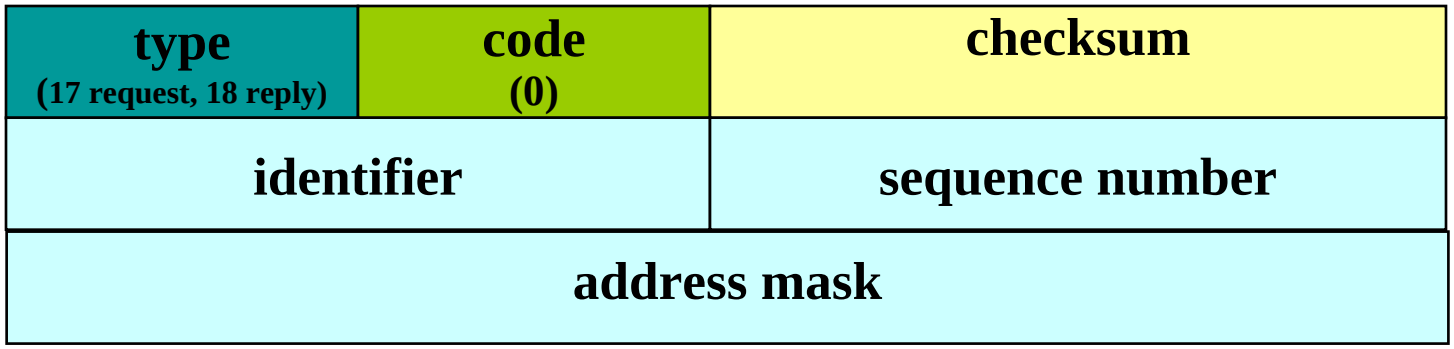


Timestamp request e reply

type (13 request, 14 reply)	code (0)	checksum
identifier		sequence number
originate timestamp		
receive timestamp		
transmit timestamp		

- Questo messaggio viene usato per scambiarsi informazioni sul clock di sorgente e destinazione
- *originate timestamp:* viene riempito dalla sorgente
- *receive timestamp*: viene riempito dalla destinazione appena ricevuto il pacchetto
- *transmit timestamp:* viene riempito dalla destinazione immediatamente prima di inviare il pacchetto di risposta

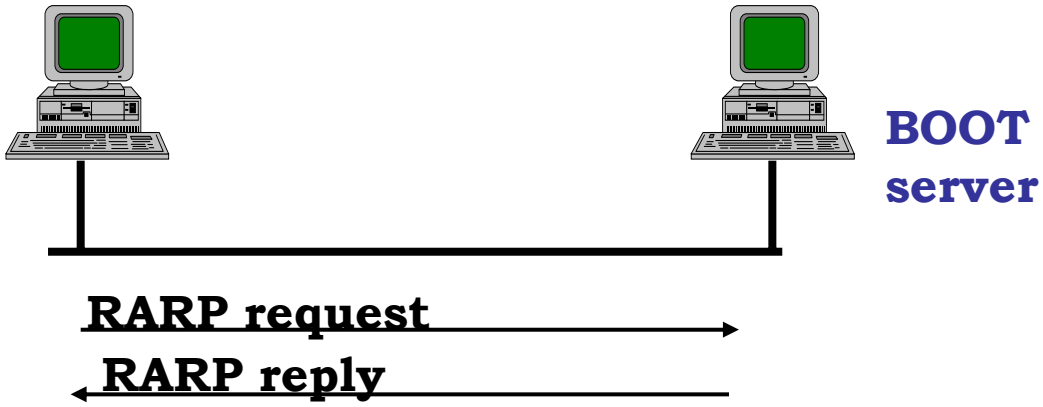
Address mask request e reply (RFC 950)



- Questo messaggio viene usato per conoscere la netmask di un host/router
- All'atto di fare booting, la macchina invia in broadcast un "Address mask request". Un gateway (o un host che agisce in vece di gateway) risponde con un "Address mask reply", comunicando tale netmask.
- Il campo address mask viene riempito da chi invia la risposta

RARP (Reverse ARP)

- Il protocollo ARP consente di associare ad un indirizzo IP noto un indirizzo fisico non noto usando la capacità di broadcast della rete sottostante
- il protocollo RARP (Reverse ARP) è in grado di effettuare l'operazione inversa:
  - un host che conosce il proprio indirizzo fisico chiede di sapere il proprio indirizzo IP
  - utile per macchine diskless che effettuano il bootstrap in rete
  - *ma non è più usato !!!*



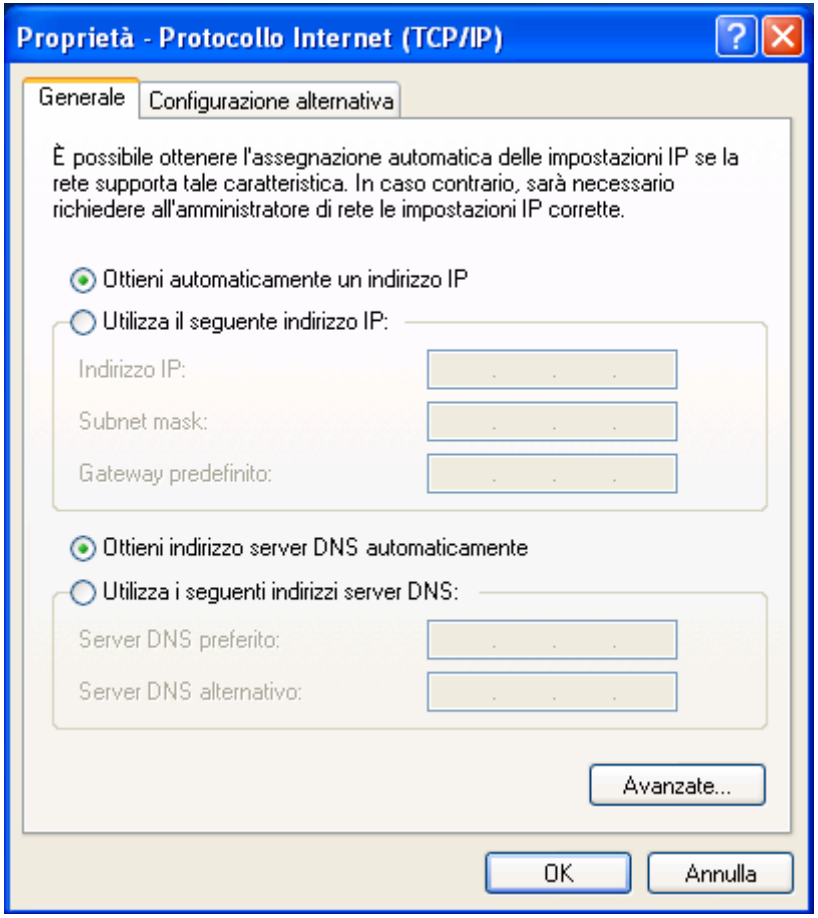
Indirizzi dinamici

- l'uso di procedure di questo tipo ha suggerito la possibilità di usare procedure per associare in modo flessibile gli indirizzi IP agli indirizzi fisici
- può essere comodo non configurare i singoli host con l'indirizzo IP, ma usare un server per memorizzare tutte le configurazioni
- in molti casi non è necessario avere un'associazione stabile tra i due indirizzi ma si può usare un'associazione dinamica (più host degli indirizzi disponibili):
  - host spesso inattivi (es. collegamenti remoti con rete d'accesso telefonica)
  - host che usano IP solo per rari scambi di informazioni

Indirizzi dinamici

- Supponiamo di avere un server in grado di fornire l'indirizzo IP ad un host su richiesta
- sono possibili diversi casi:
  - associazione statica: il server ha una tabella di corrispondenza tra indirizzi fisici e indirizzi IP e all'arrivo di una richiesta consulta la tabella e invia la risposta
  - associazione automatica: la procedura di corrispondenza nella tabella è automatizzata dal server
  - associazione dinamica: l'insieme di indirizzi IP è più piccolo del numero di host che possono usarlo

Indirizzi dinamici

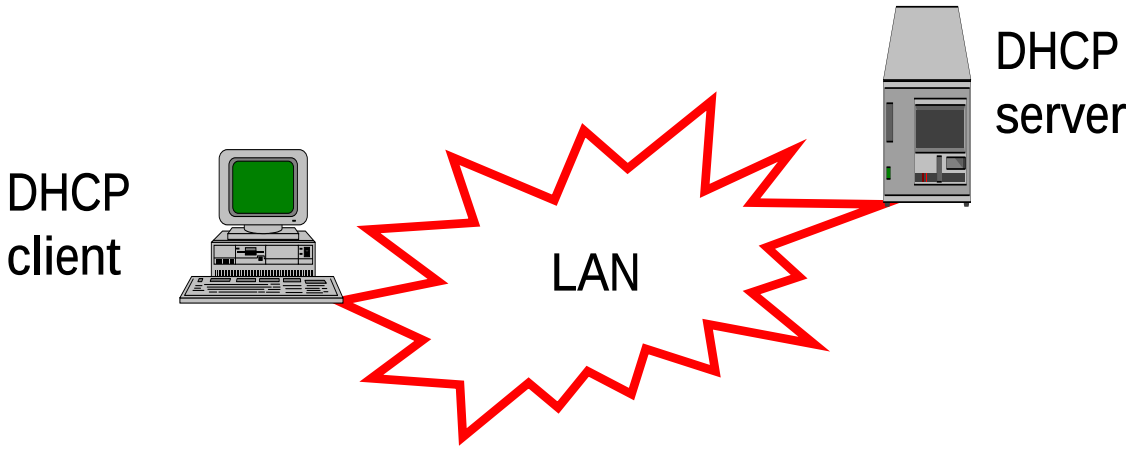


Associazione dinamica

- Il caso dell'allocazione dinamica è utile in situazioni nelle quali gli host non necessitano di avere sempre un indirizzo IP
- L'associazione deve essere temporanea (uso di timeout o procedure di rilascio esplicito)
- è possibile che all'arrivo di una richiesta non vi siano indirizzi disponili (rifiuto della richiesta)
- il dimensionamento del numero di indirizzi IP segue gli stessi principi del dimensionamento di un fascio di circuiti in telefonia

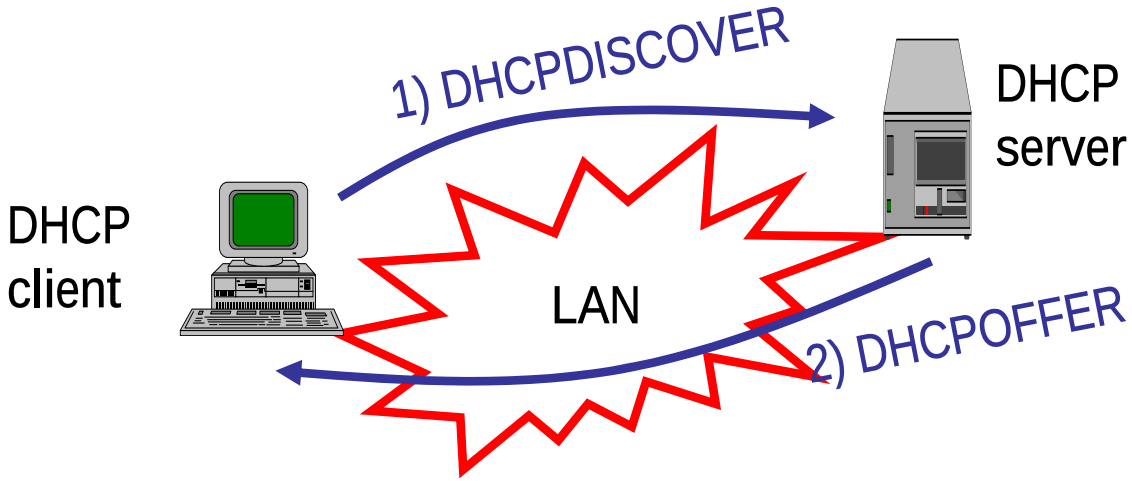
Dynamic Host Configuration Protocol (DHCP)

- per la configurazione di indirizzi IP non si usa il RARP, ma un protocollo più evoluto derivato dal BOOTP
- è un protocollo di tipo client-server



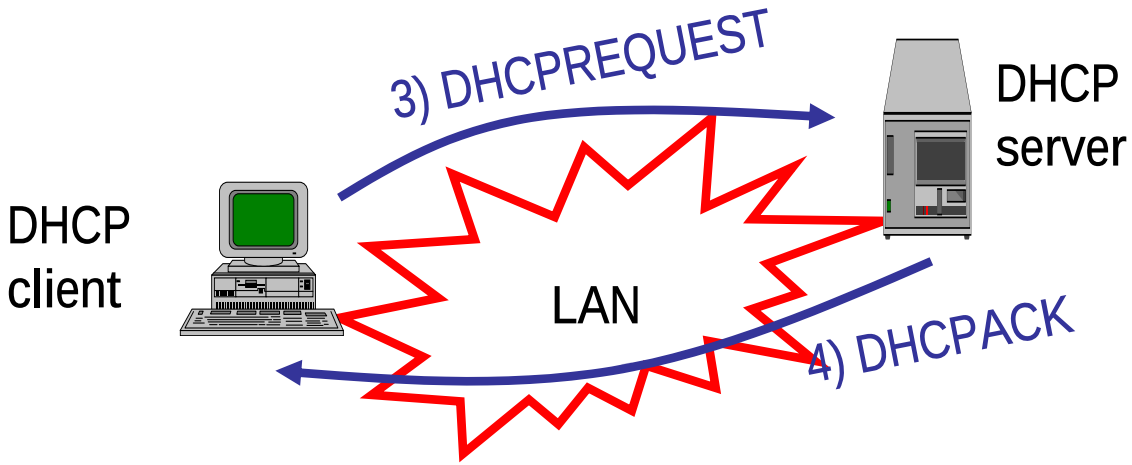
DHCP

- Un client che deve configurare il proprio stack IP invia in broadcast un messaggio di DHCPDISCOVER contenente il proprio indirizzo fisico
- Il server risponde con un messaggio di DHCPOFFER contenente un proprio identificativo e un indirizzo IP proposto



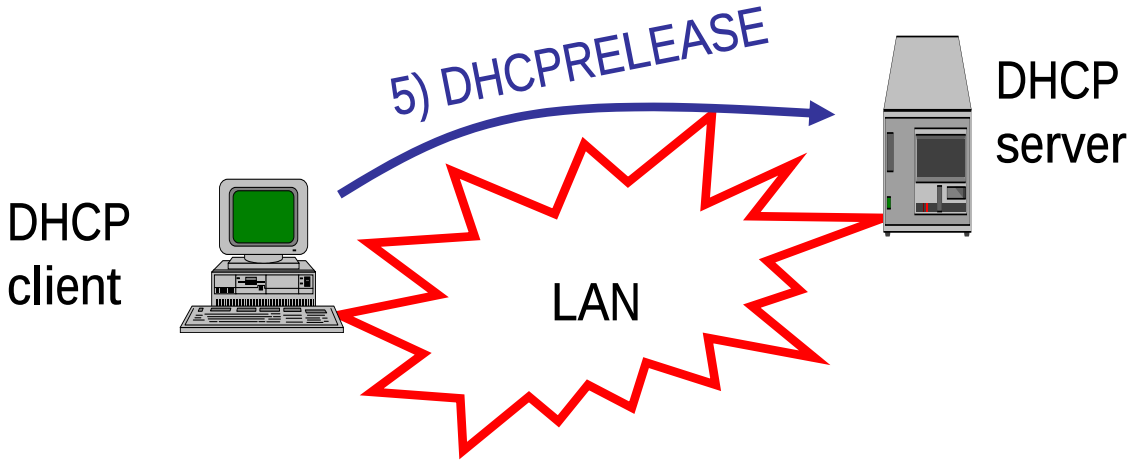
DHCP

- **Il client può accettare l'offerta inviando una DHCPREQUEST contenente l'identificativo del server (anche questo messaggio viene inviato in broadcast)**
- **Il server crea l'associazione con l'indirizzo IP e manda un messaggio di DHCPACK contenente tutte le informazioni di configurazione necessarie**



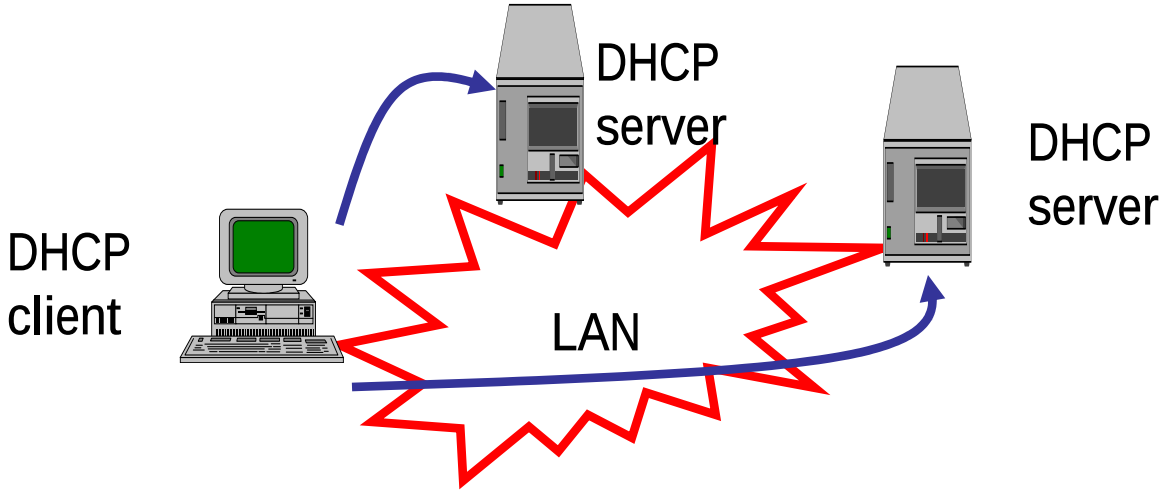
DHCP

- Parametri di configurazione
  - IP address
  - Netmask
  - Default Gateway
  - DNS server
- Il rilascio dell'indirizzo avviene con l'invio di un messaggio di DHCPRELEASE da parte del client

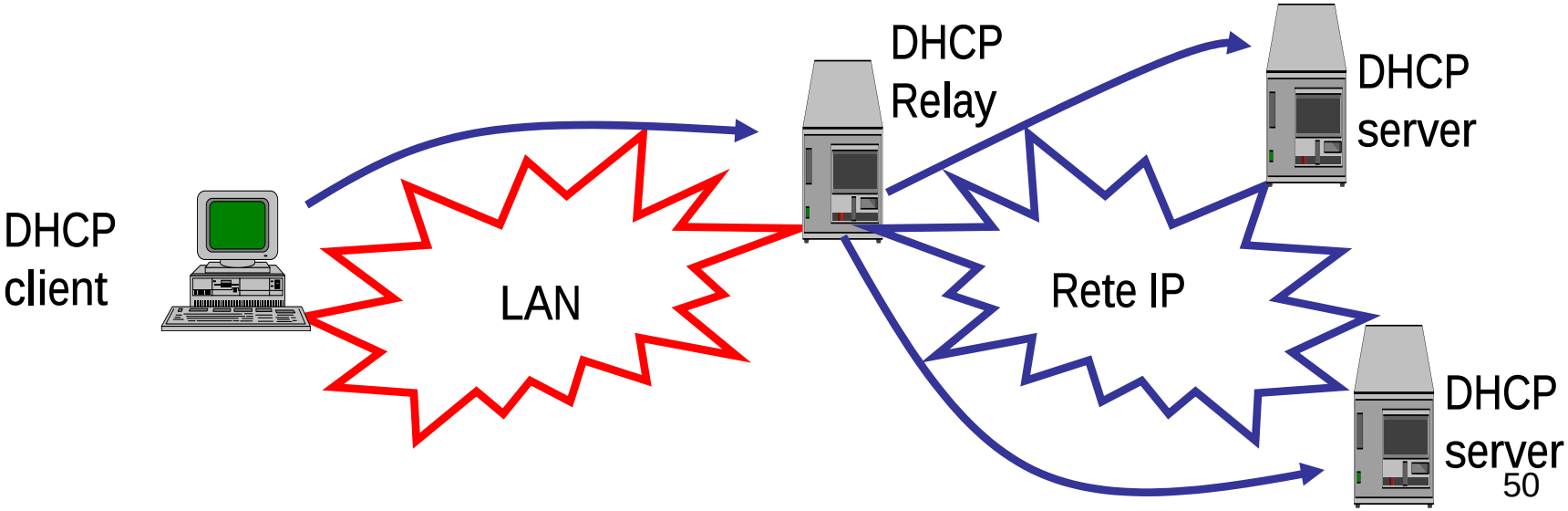


DHCP

- è possibile avere più server:

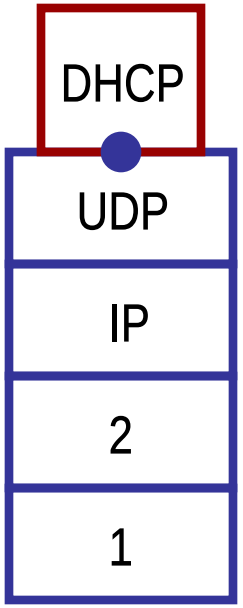


- è possibile usare dei DHCP Relay



Trasporto dei messaggi

- DHCP si appoggia su UDP per il trasporto dei messaggi
- I messaggi dei client fino all'assegnamento dell'indirizzo IP hanno:
  - ind. IP di sorgente: 0.0.0.0
  - ind. IP di destinazione: 255.255.255.255
  - porta UDP sorgente: 68
  - porta UDP destinazione: 67



Messaggi

OP	HTYPE	HLEN	HOPS
XID (Transaction ID)			
SECS		FLAGS	
CIADDR			
YIADDR			
SIADDR			
GIADDR			
CHADDR			
SNAME			
FILE			
OPTIONS			

CAMPO	BYTE	DESCRIZIONE
op	1	Tipo di messaggio (1 = BOOTREQUEST, 2 = BOOTREPLY)
htype	1	Tipo di indirizzo fisico (1 = Eth 10Mb)
hlen	1	Lunghezza ind. fisico ('6' per Eth 10Mb)
hops	1	Settato dal client a 0 e incrementato dai relay agests
xid	4	Numero casuale settato dal client e usato per evitare ambiguit\`a
secs	2	Settato dal client, numero di sec dall'inizio della procedura
flags	2	Flags (si usa solo il primo bit per chiedere una risposta multicast o unicast).
ciaddr	4	Indirizzo IP del client (settato dal client, zero se non noto)
yiaddr	4	Indirizzo IP del client (settato dal server)
siaddr	4	Indirizzo IP del server
giaddr	4	Indirizzo del relay agent
chaddr	16	Indirizzo fisico del client
sname	64	Stringa Nome del server (opzionale)
file	128	Stringa nome del file di boot (opzionale)
options	312	Lista di opzioni per il trasferimento di altre informazioni